AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions, and listings of claims in the application:

Claims 1-2 (canceled)

Claim 3 (previously presented): A method for verifying the validity of an encrypted code generated in base L, the method comprising:

obtaining an encrypted code from a user, that when decrypted is determinable to indicate a value-in base L, the encrypted code obtained by appending a third string, which is an output of applying an encryption algorithm employing a second secret code to a second string composed of an n-bit raw number and an m-bit validation number, the m-bit validation number generated by hashing, with a hash function, a first string with a first secret code, the first string composed of the n-bit raw number and the first secret code;

converting the encrypted code to a base 2 string;

decrypting the base 2 string using the second secret code to generate a recovered second string;

hashing, with the first secret code, an n-bit portion of the recovered second string concatenated with the first secret code to generate a second m-bit validation number;

comparing the remaining m-bits of the recovered second string with the second m-bit validation number to verify the validity of the encrypted code; and

if valid then crediting the user with the value indicated by the decrypted code, wherein n=32 and m=16.

Claims 4-20 (canceled)

Claim 21 (previously presented): A method for awarding incentive points to a user, comprising the steps of:

receiving on-line from a user a code generated with encrypted information and obtained by the user off-line;

verifying the validity of the code by processing the encrypted information; and awarding incentive points to the user if the code is valid,

wherein the code is generated by:

providing a number portion,

deriving a validation portion from the number portion,

appending the validation portion to the number portion to form a string,

encrypting the string, and

deriving the code from the encrypted string by converting the encrypted string to base L string,

wherein the string is 48-bits long and the number portion is 32-bits long.

Claims 22-24 (canceled)

Claim 25 (previously presented): A computer-enabled method for awarding incentive points to a user, comprising:

receiving on-line from the user a code generated with encrypted information and obtained by the user offline;

verifying the validity of the code by processing the encrypted information; and

awarding incentive points to the user if the code is valid, wherein the code is generated by:

providing a number portion, Sl_{INT}, from a first string, S1

arranging a first secret key, K1, next to the number portion, Sl_{INT}, from S1, to form a second string, S2,

applying a hash function to S2 to produce a third string, S3,

extracting a validation portion, $S1_{VAL}$, from S3 and arranging $S1_{VAL}$ next to Sl_{INT} in S1 ($S1 = S1_{VAL}, + Sl_{INT}$),

encrypting S1 using a second secret key, K2, to form a fourth string, S4, and

deriving the code by converting S4 to a base L fixed-length code string,

wherein S1 is 48-bits long and the number portion, Sl_{INT}, is 32-bits long.

Claims 26-32 (canceled)

Claim 33 (previously presented): A method for offline-online management of incentive points, comprising:

receiving a code, generated by providing a number portion, deriving a validation portion from the number portion, appending the validation portion to the number portion to form a string, encrypting the string, and deriving the code from the encrypted string by converting the encrypted string to base L string, the code obtained off-line and received on-line; and

submitting the code to a server that has valid codes, wherein the code is associated with N points maintained by the server in a user account, wherein each point, characterized as a purchase or attention incentive point, is redeemable; and

verifying the code against the valid codes to determine if it is valid, wherein if the code is valid, a predetermined number of points are added to the user account,

wherein the string is 48-bits long and the number portion is 32-bits long.

Claims 34-36 (canceled)

Claim 37 (previously presented): A method for generating a code that corresponds to incentive points, comprising:

providing a number portion, Sl_{INT};

arranging a first secret key, K1, next to Sl_{INT}~~;~~ to form a second string, S2,

applying a hash function to S2 to produce a third string, S3, extracting a validation portion, $S1_{VAL}$, from S3 and arranging $S1_{VAL}$, next to Sl_{INT} to produce] S1 ($S1 = Sl_{VAL}+ Sl_{INT}$),

encrypting S1 using a second secret key, K2, to form a fourth string, S4, and

deriving the code by converting S4 to a base L fixed-length code string; and

fixing the code onto a medium such that the code is obtainable from the medium off-line,

wherein S1 is 48-bits long and the number portion, Sl_{INT}, is 32-bits long.

Claim 38 (canceled)

Claim 39 (currently amended): ~~A method as in claim 38,~~

A method for offline-online management of incentive points, comprising:

receiving a code, generated by providing a number portion, deriving a validation portion from the number portion, appending the validation portion to the number portion to form a string, encrypting the string, and deriving the code from the encrypted string by converting the encrypted string to a base L string, the code obtained off-line and received on-line;

submitting the code to a server that has valid codes, wherein the code is associated with N points maintained by the server in a user account, wherein each point, characterized as a purchase or attention incentive point, is redeemable;

verifying the code against the valid codes to determine if it is valid, wherein if the code is valid, a predetermined number of points are added to the user account,

wherein the validation portion is at least 16 bits long,

wherein the step of verifying the code includes:

converting the code from a base L string into a base 2 string, $S4_{BASE2}$,

decrypting $S4_{BASE2}$ using a second secret key, K2, to form a decrypted first string, S1',

providing a number portion from S',

arranging a first secret key, K1, next to the number portion from S1' to form a second string, S2',

applying a hash function to S2' to form a third string S3',

extracting a validation portion from S3' and a validation portion from S1', and

determining if the code is valid by comparing the validation portion from S3' with the validation portion from S1',

wherein S3' is at least 16 bits long, and

wherein S1' is 48-bits long and the number portion is 32-bits long.

Claim 40 (canceled)